**Міністерство освіти і науки України**

**ЧЕРКАСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ**

**ім. Богдана Хмельницького**

**Факультет** Обчислювальної техніки, інтелектуальних та управляючих систем

**Кафедра** Програмного забезпечення автоматизованих систем

# ЛАБОРАТОРНА РОБОТА № 1

з дисципліни «Вступ до інженерії програмного забезпечення»

**Тема:** Організація програми для підрахунку суми цілих чисел у бінарному вигляді.  **Варіант 7 mod 5 = 2**

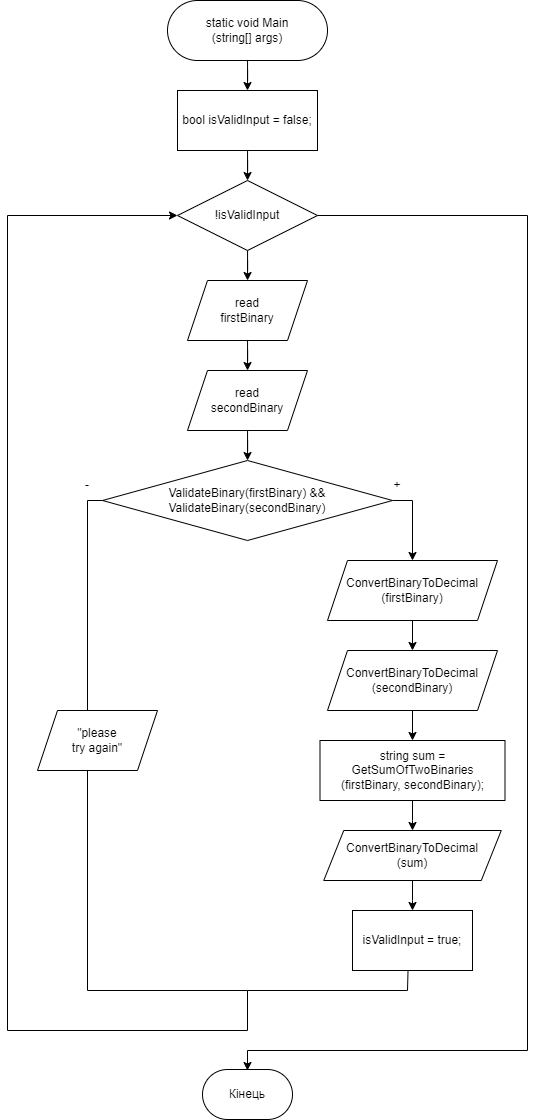
**Виконав:** студент гр. КС-231   
Киба Д.В.

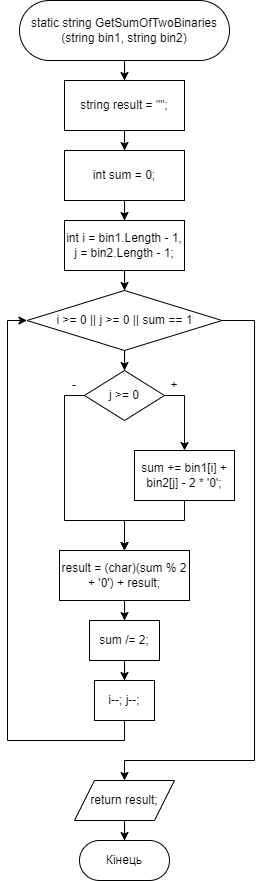
Черкаси, 2023

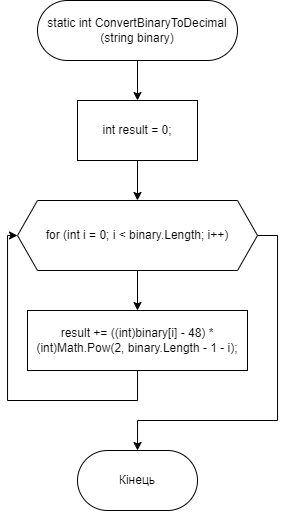
# Постановка завдання :

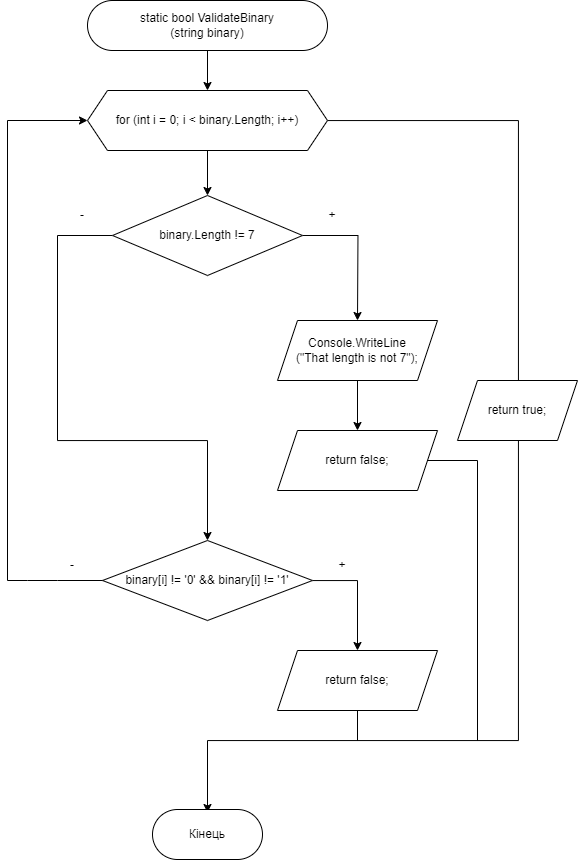
Написати програму яка виконує додавання двох чисел введених з клавіатури у бінарному вигляді, а виводить на екран результат додавання (у бінарному вигляді) та введені числа і їх суму у десятковій формі.

# Алгоритм вирішення завдання (блок-схеми):

  
Рис. 1. Метод Main

  
Рис. 2. Метод GetSumOfTwoBinaries

  
Рис. 3. Метод ConvertBinaryToDecimal

  
Рис. 4. Метод ValidateBinary

* 1. **Текст програми для вирішення завдань (лістинг):**

using System;

using System.Text;

namespace BinarySummator

{

class Program

{

static void Main(string[] args)

{

Console.OutputEncoding = UTF8Encoding.UTF8;

Console.WriteLine("Introduction to Software Engineering, Labwork #1\nVariant 7 mod 5 = 2\nLength of combination - 7 symbols.\n");

bool isValidInput = false;

while (!isValidInput)

{

Console.Write("Enter the first binary number : ");

string firstBinary = Console.ReadLine();

Console.Write("Enter the second binary number: ");

string secondBinary = Console.ReadLine();

Console.WriteLine();

if (ValidateBinary(firstBinary) && ValidateBinary(secondBinary))

{

Console.WriteLine($"The first binary number in decimal representation: {ConvertBinaryToDecimal(firstBinary)}");

Console.WriteLine($"The second binary number in decimal representation: {ConvertBinaryToDecimal(secondBinary)}");

string sum = GetSumOfTwoBinaries(firstBinary, secondBinary);

Console.WriteLine($"The sum of two binary numbers is {sum}\nThe decimal representation of the sum is {ConvertBinaryToDecimal(sum)}");

isValidInput = true;

}

else

{

Console.WriteLine("The provided input is incorrect, please try again.");

}

}

}

static string GetSumOfTwoBinaries(string bin1, string bin2)

{

string result = "";

int sum = 0;

int i = bin1.Length - 1, j = bin2.Length - 1;

while (i >= 0 || j >= 0 || sum == 1)

{

if (j >= 0)

{

sum += bin1[i] + bin2[j] - 2 \* '0';

}

result = (char)(sum % 2 + '0') + result;

sum /= 2;

i--; j--;

}

return result;

}

static int ConvertBinaryToDecimal(string binary)

{

int result = 0;

for (int i = 0; i < binary.Length; i++)

{

result += ((int)binary[i] - 48) \* (int)Math.Pow(2, binary.Length - 1 - i);

}

return result;

}

static bool ValidateBinary(string binary)

{

for (int i = 0; i < binary.Length; i++)

{

if (binary.Length != 7) //hardcode

{

Console.WriteLine("That length is not 7");

return false;

}

if (binary[i] != '0' && binary[i] != '1')

{

return false;

}

}

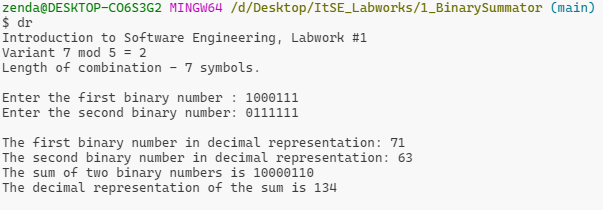
return true;

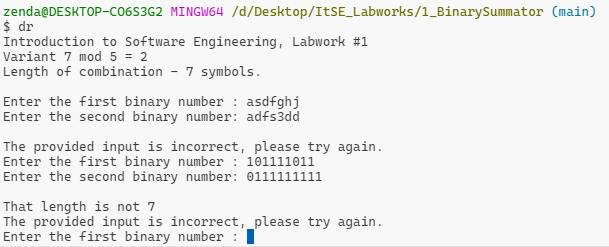
}

}

}

}

* 1. **Результат виконання програми:**

****

Валідація працює.

# Висновок:

Лабораторна робота реалізує програму для обчислення суми двох введених користувачем двійкових чисел. Користувачу пропонується ввести два двійкових числа, після чого програма перевіряє їх правильність та конвертує у десятковий формат. Після цього виконується операція додавання двох чисел у двійковій формі, результат також конвертується у десятковий формат та виводиться на екран.

Програма використовує цикл для перевірки коректності введених користувачем даних та забезпечує повторний ввід у разі помилки. Також вона містить функції для переведення двійкового числа у десяткову систему та для самого додавання двох двійкових чисел.

Для перевірки коректності введення двійкових чисел реалізовано функцію ValidateBinary, яка перевіряє правильність введення та довжину введеного числа (у цьому випадку довжина повинна дорівнювати 7 символам).

Програма написана мовою C# з використанням стандартної бібліотеки .NET. Алгоритм додавання двох двійкових чисел використовує підхід сумування по позиціях та переносів.